MA40090 2015/16 Fake exam solutions.

1 Question 1

Part a

Agglomerative hierarchical clustering is a method for producing nested clusterings. It proceeds as follows:

1. Begin with all points in individual clusters
2. While the number of clusters is bigger than 1 do
   (a) Find the two clusters that are closest together in the sense that the linkage between them is minimal
   (b) Join these two clusters, noting the value of the linkage, which is used as the height of the join on the dendrogram

The three most common types of linkage are as follows:

- Single linkage $d(A, B) = \min_{x\in A, y\in B} \|x - y\|$
- Complete linkage $d(A, B) = \max_{x\in A, y\in B} \|x - y\|$
- Average linkage $d(A, B) = |A|^{-1}|B|^{-1} \sum_{x\in A, y\in B} \|x - y\|$

The clustering found by cutting the dendrogram at β can be interpreted as

- Single linkage: For any point in cluster A there is another point in the same cluster that is at most β units away.

- Complete linkage: For any point in cluster A, every point in the same cluster that is at most β units away.
- Average linkage: There is no interpretation for the cut height.

Part b

k-means clustering has the following steps:

1. Randomly assign k points as cluster centres
2. Repeat until convergence:
   (a) Update $\ell(\cdot)$ to assign each point to the cluster associated with the closest (in Euclidian distance) cluster centre (breaking ties consistently)

   (b) Choose new cluster centres as the centroid of the current cluster, i.e.
   $$z_i^{\text{new}} = \frac{1}{|C_i|} \sum_{x \in C_i} x,$$
   where $C_i = \{x \in X : \ell(x) = i\}$.

The proof proceeds as follows:

- Assume that steps 2.1 and 2.2 of the algorithm never increase the cost function. (we will get to this)
- The assignment function $\ell(\cdot)$ and the centres $\{z_i\}_{i=1}^{k}$ can only take a finite number of values (only a finite number of partitions of the data, every z_i must be a mean of a subset of the data)
- The cost function is bounded below by zero
- Ties are broken consistently, so the algorithm cannot oscillate.
- Therefore, the algorithm can only take a finite number of non-decreasing steps before terminating at a local minimum and if it gives the same value twice it is necessarily generated by the same clustering.

The cost function after step 2.1 satisfies

$$\begin{aligned} cost &= \sum_{i=1}^{k} \sum_{x \in C_i} \|x - z_i\|^2 \\ &= \sum_{i=1}^{k} \sum_{x \in \{\ell(x)=i\}} \|x - z_i\|^2 \\ &= \sum_{x \in X} \min_{i \in \{1,\ldots,k\}} \|x - z_i\|^2 \qquad (1) \\ &\leq cost(C_1', \ldots, C_k', z_1, \ldots, z_k) \end{aligned}$$

for any partition $\{C_i'\}_{i=1}^k$ of X. The equality (1) reflects that step 2.1 chooses the closest centre and hence minimises the distance.

To see that step 2.2 does not increase the cost function, note that as 2.2 does not change $\ell(\cdot)$, we can treat it as fixed. The cost function (dropping the dependence on the partition)

$$cost(z_1, \ldots, z_k) = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - z_i\|^2$$

is quadratic and its gradient is

$$\nabla_{z_i} c = -2 \sum_{x \in C_i} (x - z_i).$$

Hence the global minimum (+ve second derivative) is found at

$$z_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$$

and $cost(z_1^{\text{new}}, \ldots, z_k^{\text{new}}) \leq cost(z_1', \ldots, z_k')$ for any other set of centres $\{z_i'\}_{i=1}^k$ and hence the objective does not increase.

1.1 Part c

One possible solution occurs when the two clusters form concentric circles, where the distance between two points in a cluster is always smaller than the distance between the rings. For full marks you need to point this out.

2 Question 2

Part a

The first principle component is the unit vector $\boldsymbol{v}_1$ in which the sample variance of $\boldsymbol{x}^T\boldsymbol{v}$ is maximised. Assume the data is centred, then the sample variance of $\boldsymbol{x}^T\boldsymbol{v}$ is

$$\frac{1}{n-1}\sum_{i=1}^{n}\boldsymbol{v}_i^T\boldsymbol{x}\boldsymbol{x}_i^T\boldsymbol{v} = \|\boldsymbol{X}\boldsymbol{v}\|_2^2,$$

where $\boldsymbol{X}$ is the $n\times p$ matrix with the ith row containing $\boldsymbol{x}_i^T$. Hence the first principle component maximises $\|\boldsymbol{X}\boldsymbol{v}\|_2$ over all unit vectors $\boldsymbol{v}$, which means that $\boldsymbol{v}_1$ is the eigenvector corresponding to the first eigenvalue of the sample covariance matrix $\hat{\boldsymbol{\Sigma}} = (n-1)^{-1}\boldsymbol{X}^T\boldsymbol{X}$.

The second principle component is the unit vector $\boldsymbol{v}_2$ that is orthogonal to $\boldsymbol{v}_1$ that maximises the sample of variance of $\boldsymbol{x}^T\boldsymbol{v}$. This can be interpreted as maximising the variance over all directions that are "independent" of $\boldsymbol{v}_1$. It is given by the second eigenvector of the sample covariance matrix.

Part b

The total sample variance for the data set $\boldsymbol{X}$ is defined as $\mathrm{tr}(\hat{\boldsymbol{\sigma}}) = \sum_{i=1}^{n}\sigma_i^2$ by definition. The sample variance in the direction of the jth principal component v_j is $\|\boldsymbol{X}\boldsymbol{v}_j\|_2^2 = \lambda_j$ using the results in part a.

Part c

For any clustering $\{C_1,\ldots,C_k\}$, let $\boldsymbol{A}\in\mathbb{R}^{n\times k}$ be the **cluster indicator matrix** with

$$A_{ij} = \begin{cases} |C_j|^{-1/2}, & \boldsymbol{x}_i \in C_j \\ 0, & \text{otherwise} \end{cases}.$$

The ith row of $\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{X}$ is the cluster centre $\boldsymbol{x}_i$ is assigned to and hence, the k-means cost function can be written as

$$\sum_{i=1}^{k}\sum_{x\in C_i}\|\boldsymbol{x}-\boldsymbol{z}_i\|_2^2 = \left\|\boldsymbol{X}-\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{X}\right\|_F^2.$$

On the other hand, we recall that if we write $\boldsymbol{X}_k = \boldsymbol{U}_k \boldsymbol{\Sigma}_k \boldsymbol{V}_k^T$, where the k subscript indicates that only the first k singular values/vectors are used, then

$$\boldsymbol{X}_k = \arg \min_{\text{rank}(\boldsymbol{B})=k} \|\boldsymbol{X} - \boldsymbol{B}\|_2 = \arg \min_{\text{rank}(\boldsymbol{B})=k} \|\boldsymbol{X} - \boldsymbol{B}\|_F .$$

Hence both problems minimise the same objective function. The difference is that while PCA minimises over the full set of all orthogonal matrices, k-means optimises over a discrete subset of cluster indicator matrices.

Part d

We replace the features $\boldsymbol{x}$ with the extended features $\Phi(\boldsymbol{x})$ for some feature map $\Phi(\cdot)$. We can then proceed as normal. Assume that the data is centred on the extended feature space (i.e. $\sum_{i=1}^n \Phi(\boldsymbol{x}_i) = 0$) and define $\hat{\Sigma} = (n-1)^{-1} \sum_{i=1}^n \Phi(\boldsymbol{x}_i)\Phi(\boldsymbol{x}_i)^T$ and then the first principal component is

$$v = \arg\max_v \frac{v^T \hat{\Sigma} v}{v^T v}.$$

This is equivalent to finding the largest (λ, v) such that $(n-1)^{-1} \sum_{i=1}^n \Phi(\boldsymbol{x}_i)\Phi(\boldsymbol{x}_i)^T v = \lambda v$. Examining this equation, it's clear that we can find real numbers $\{\alpha_i\}_{j=1}^n$ such that $v = \sum_{j=1}^n \alpha_j \Phi(\boldsymbol{x}_j)$. Hence, we want to find $(\lambda, \boldsymbol{\alpha})$ such that

$$(n-1)^{-1} \sum_{i=1}^n \sum_{j=1}^n \alpha_j \left(\Phi(\boldsymbol{x}_i)^T \Phi(\boldsymbol{x}_j)\right) \Phi(\boldsymbol{x}_i) = \lambda \sum_{j=1}^n \alpha_j \Phi(\boldsymbol{x}_j)$$

$$(n-1)^{-1} \sum_{i=1}^n \sum_{j=1}^n \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \Phi(\boldsymbol{x}_i) = \lambda \sum_{j=1}^n \alpha_j \Phi(\boldsymbol{x}_j).$$

For any $\boldsymbol{x}_k$ in the training set, we multiply the previous equation on the left by $\Phi(\boldsymbol{x}_k)^T$ and get

$$(n-1)^{-1} \sum_{i=1}^n \sum_{j=1}^n \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) K(\boldsymbol{x}_i, \boldsymbol{x}_k) = \lambda \sum_{j=1}^n \alpha_j K(\boldsymbol{x}_j, \boldsymbol{x}_k).$$

Examining this equation closely, and defining the kernel matrix $\boldsymbol{K}$ by $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, this is equivalent to

$$(n-1)^{-1} \boldsymbol{K}^2 \boldsymbol{\alpha} = \lambda \boldsymbol{K} \boldsymbol{\alpha}.$$

This means that we can find the first kernelised principal component by finding the largest eigenpair of the Kernel matrix $\boldsymbol{K}$.

The RKHS in this case contains the set of all possible principal components.

3 Question 3

Part a

It is enough to show that for any other classifier h,

$$\Pr(Y \neq h(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) - \Pr(Y \neq h^*(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) \geq 0.$$

Now,

$$\begin{aligned}
\Pr(Y \neq h(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) &= 1 - \Pr(Y = h(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) \\
&= 1 - \sum_{k=0}^{1} \Pr(Y = k, h(\boldsymbol{X}) = k \mid \boldsymbol{X} = \boldsymbol{x}) \\
&= 1 - \sum_{k=0}^{1} \Pr\left(Y = k \mid \boldsymbol{X} = \boldsymbol{x}, h(\boldsymbol{X} = k)\right) \Pr(h(\boldsymbol{X}) = k \mid \boldsymbol{X} = \boldsymbol{x}).
\end{aligned}$$

Under the event $\{\boldsymbol{X} = \boldsymbol{x}\}$, $h(\boldsymbol{X}) = h(\boldsymbol{x})$ is a deterministic function, so $\Pr(h(\boldsymbol{X}) = k \mid \boldsymbol{X} = \boldsymbol{x}) = 1$ or 0 which implies the events $\{h(\boldsymbol{X}) = k \mid \boldsymbol{X} = \boldsymbol{x}\}$ and $\{Y = k \mid \boldsymbol{X} = \boldsymbol{x}\}$ are conditionally independent (by definition of independence!). Furthermore, $\Pr(h(\boldsymbol{X}) = 1 \mid \boldsymbol{X} = \boldsymbol{x}) = h(\boldsymbol{x})$. Therefore

$$\Pr(Y \neq h(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) = 1 - \left[m(\boldsymbol{x})h(\boldsymbol{x}) + (1 - h(\boldsymbol{x}))(1 - m(\boldsymbol{x}))\right].$$

Hence

$$\begin{aligned}
&\Pr(Y \neq h(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) - \Pr(Y \neq h^*(\boldsymbol{X}) \mid \boldsymbol{X} = \boldsymbol{x}) \\
&\quad = -\left[m(\boldsymbol{x})h^*(\boldsymbol{x}) + (1 - h^*(\boldsymbol{x}))(1 - m(\boldsymbol{x}))\right] \\
&\qquad - \left[m(\boldsymbol{x})h(\boldsymbol{x}) + (1 - h(\boldsymbol{x}))(1 - m(\boldsymbol{x}))\right] \\
&\quad = 2\left(m(\boldsymbol{x}) - \frac{1}{2}\right)(h^*(\boldsymbol{x}) - h(\boldsymbol{x})).
\end{aligned}$$

When $m(\boldsymbol{x}) > 1/2$, $h^*(\boldsymbol{x}) = 1$ and so both terms are non-negative. Hence the result.

Part b

Linear discriminant analysis and logistic regression have three main differences. The first is that LDA is based on a full generative model for the features and labels, that is it posits a joint distribution $F_{\boldsymbol{x},y}$, whereas logistic regression only models the required conditional distribution $\pi(y = k \mid \boldsymbol{X} = \boldsymbol{x})$. The second is that when the data is linearly separable, the estimates of the coefficients in logistic regression explode to infinity, while LDA remains well behaved. The final difference is that LDA can be computed using explicit formulas for each of the terms, while logistic regression requires numerical optimisation of the log-likelihood function.

Part c

The likelihood is given by

$$\begin{aligned} L(\beta_0, \boldsymbol{\beta}) &= \prod i = 1^n p(\boldsymbol{x}_i)^{y_i)} (1 - p(\boldsymbol{x}_i))^{1-y_i} \\ &= \prod_{i=1}^{n} \left(\frac{e^{\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 + e^{\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}}} \right)^{y_i} \left(\frac{1}{1 + e^{\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}}} \right)^{1-y_i} \\ &= \prod_{i=1}^{n} \frac{e^{y_i(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta})}}{1 + e^{\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}}}. \end{aligned}$$

Assume that $(\beta_0, \boldsymbol{\beta})$ are such that the decision boundary $\beta_0 + \boldsymbol{x}^T \boldsymbol{\beta} = 0$ perfectly separates the data Then it follows that, for any $c > 0$, the parameters $(c\beta_0, c\boldsymbol{\beta})$ will also perfectly separate the data. When $y_i = 1$, by assumption, $t_i(c) = c(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}) > 0$ and hence its contribution to the likelihood

$$\frac{e^{t_i(c)}}{1 + e^{t_i(c)}} = 1 - \frac{1}{1 + e^{t_i(c)}}$$

is an increasing function of c. Similarly, when $y_i = 0$, by assumption, $t_i(c) = c(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}) < 0$ and hence its contribution to the likelihood

$$\frac{1}{1 + e^{t_i(c)}}$$

is an increasing function of c. Hence there is no maximum likelihood estimator for $(\beta_0, \boldsymbol{\beta})$ when the data is linearly separable.

Part d

For any classification function h, the expected loss can be is given directly as

$$\begin{aligned}
R(h) &= \alpha_{FN} \Pr(h(x) = 0, y = 1 \mid X = x) + \alpha_{FP} \Pr(h(x) = 1, y = 0 \mid X = x) \\
&= \alpha_{FN} \Pr(y = 1 \mid X = x, h(x) = 0) \Pr(h(x) = 0 \mid X = x) \\
&\qquad + \alpha_{FP}(1 - \Pr(y = 1 \mid X = x, h(x) = 1)) \Pr(h(x) = 1 \mid X = x).
\end{aligned}$$

Following the solution to Part (a) and writing $m(x) = \Pr(y = 1 \mid X = x)$, we can write this as

$$R(h) = \alpha_{FN} m(x)(1 - h(x)) + \alpha_{FP}(1 - m(x))h(x).$$

Let h^* be the optimal classifier. Then for any other rule h,

$$\begin{aligned}
R(h^*) - R(h) &= -\alpha_{FN} m(x)h^*(x) + \alpha_{FP} h^*(x) - \alpha_{FP} m(x)h^*(x) \\
&\qquad + \alpha_{FN} m(x)h(x) - \alpha_{FP} h(x) + \alpha_{FP} m(x)h(x) \\
&= (h^*(x) - h(x)) \left[\alpha_{FP} - (\alpha_{FN} + \alpha_{FP})m(x)\right] \\
&= (\alpha_{FN} + \alpha_{FP})(h^*(x) - h(x)) \left(\frac{\alpha_{FP}}{\alpha_{FN} + \alpha_{FP}} - m(x)\right)
\end{aligned}$$

If h^* is optimal, then $R(h^*) - R(h) \leq 0$. Clearly, the classifier

$$h^*(x) = \begin{cases} 1, & m(x) > \frac{\alpha_{FP}}{\alpha_{FN}} \\ 0, & \text{otherwise} \end{cases}$$

satisfies this requirement.

4 Question 4

Part a

The support vector classifier can be written as

$$\min_{\beta_0, \boldsymbol{\beta}, \boldsymbol{\xi}} \frac{1}{2} \|\boldsymbol{\beta}\|^2 + C \sum_{i=1}^{n} \xi_i$$

Subject to:

$$y_i(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta}) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

To derive the dual form, we first need to form the Lagrangian

$$L(\beta_0, \boldsymbol{\beta}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{1}{2}\left\|\boldsymbol{\beta}\right\|^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i\left(1 - \xi_i - y_i(\beta_0 + \boldsymbol{x}_i^T\boldsymbol{\beta})\right) - \sum_{i=1}^{n}\mu_i\xi_i.$$

We derive the dual problem by maximising the Lagrangian with respect to the primal variables β_0, β_0 and $\boldsymbol{\xi}$.

Maximising w.r.t. β_0 Solving $\frac{\partial L}{\partial \beta_0} = 0$ yields $\sum_{i=1}^{n}\alpha_i y_i = 0$.

Maximising w.r.t. $\boldsymbol{\beta}$ Solving $\nabla_{\boldsymbol{\beta}} L = \mathbf{0}$ we get

$$\frac{\partial L}{\partial \beta_j} = \beta_j - \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i = 0$$

and hence $\boldsymbol{\beta} = \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i$.

Maximising w.r.t. $\boldsymbol{\xi}$ Solving $\nabla_{\boldsymbol{\xi}} L = \mathbf{0}$ we get

$$\frac{\partial L}{\partial \xi_j} = C - \alpha_j - \mu_j = 0$$

which implies that $\boldsymbol{\mu} = C - \boldsymbol{\alpha}$.

The dual Lagrangian Substituting these in we get

$$\begin{aligned}
L_D(\boldsymbol{\alpha}, \boldsymbol{\mu}) &= \max_{\boldsymbol{\xi}, \boldsymbol{\beta}, \beta_0} L(\beta_0, \boldsymbol{\beta}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \\
&= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i - \sum_{i=1^n}\alpha_i\xi_i - \beta_0\sum_{i=1}^{n}\alpha_i y_i \\
&\quad - \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j - C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i\xi_i \\
&= \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j
\end{aligned}$$

Complementary slackness As the primal constraints are linear, this problem does not have a duality gap (i.e. the minimum value obtained by the primal problem is the same as the maximum value obtained by the dual problem) and hence we have complementary slackness conditions at the optimal points

$$\begin{aligned} \alpha_i \left(1 - \xi_i - y_i(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta})\right) &= 0 \\ \mu_i \xi_i &= 0. \end{aligned}$$

The first condition says that when the strict form of the primal constraint $y_i(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta} > 1 - \xi_i$ is satisfied, then α_i must be zero. Only when the point $\boldsymbol{x}_i$ either lies on or on the wrong side of the separating hyperplane, can we have $\alpha_i \geq 0$. The second condition implies $(C - \alpha_i)\xi_i = 0$, so either $\xi_i = 0$, in which case the margin is not violated, or $\alpha_i = C$ when the margin is violated.

This means that only the points that are either on the separating hyperplanes or on the wrong side of them contribute to the solution of the SVC.

The dual problem The dual problem can finally be stated as

$$\max_{\boldsymbol{\lambda}} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j \boldsymbol{x}_j^T \boldsymbol{x}_i$$

Subject to:

$$\begin{aligned} &\lambda_i \left[(1 - \xi_i) - y_i(\beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta})\right] = 0, && i = 1, \ldots, n, \\ &\sum_{i=1}^{n} \lambda_i y_i = 0, && \\ &\mu_i \xi_i = 0, && i = 1, \ldots, n, \\ &0 \leq \mu_i \leq C, && i = 1, \ldots, n, \\ &\boldsymbol{\lambda} \geq \boldsymbol{0}. && \end{aligned}$$

Part b (Not examinable 2015/16)

The k-nearest neighbours method for classifying a point $\boldsymbol{x}$ into a class label in $\{1, 2, \ldots, K\}$ finds the k training points that are closest to $\boldsymbol{x}$, finds the most common label (breaking ties consistently), and assigns that label to $\boldsymbol{x}$.

Part c

A Reproducing Kernel Hilbert space $H(K)$ is a complete inner product space (i.e. a linear vector space with an inner product $\langle \cdot, \cdot \rangle_{H(K)}$ such that any sequence that converges in the norm defined through the inner product has a limit in H) containing functions defined on $\mathcal{X}$ with the property that the value of the function at any point is finite. This implies, and is implied by, the existence of a Kernel function $K(\cdot, \cdot)$, such that, for any $x \in \mathcal{X}$, $K(\cdot, x) \in H(K)$. This kernel function has the reproducing property

$$f(x) = \langle f, K(\cdot, x) \rangle_{H(K)} .$$

The support vector machines can be written as

$$\min_{f \in H(K)} \sum_{i=1}^{n} \max\{0, y_i f(\boldsymbol{x}_i)\} + \lambda \|f\|_{H(K)} ,$$

where $\lambda = (2C)^{-1}$, where C is the constant in the $C \sum_{i=1}^{n} \xi_i$ term in the primal formulation of the SVM. Hence a large C means we don't penalise big values of $\|f\|_{H(K)}$ too much and hence leads to a very wiggly function. This makes sense as a large C penalises misclassification strongly.

Part d

There are many answers to this question. An example would be the case where the Naive Bayes assumption does not hold, i.e. we cannot assume that the features and labels are drawn i.i.d from some distribution $F_{\boldsymbol{x},y}$. When this distribution changes, it is not possible to assume that the model fitted on the validation set is still a decent model of reality.